|  |  |  |  |
| --- | --- | --- | --- |
| 姓名：刘赟 | 专业：物联网工程 | 班级：20181191班 | 学号：2018119117 |
| 科目：嵌入式应用系统开发 | | 实验日期：2020/7/6 | |
| 实验题目：信号量 | | | |
| 【实验目的】   1. 在多任务操作系统中，不同的任务之间需要同步运行，信号量功能可以为用户提供这方面的支持。通过利用信号量，实现任务之间同步或临界资源的互斥访问。 | | | |
| 【实验内容】  1.实验设备   硬件：PC机一台   小熊派开发板一套   软件：XP/win7、8、8.1、10 系统，IoT Studio，git 客户端 [sourcetree](https://www.sourcetreeapp.com/)，课程使用 sourcetree 和 [github](http://www.github.com/) 作为项目管理和协作系统。  2.实验内容和要求  本实验中将创建两个任务，一个低优先级任务task1，一个高优先级任务task2，两个任务都会每隔2s在串口打印自己的任务id号，在串口终端中观察两个任务的运行情况。 | | | |
| 【实验结果】   1. 代码实现   osal\_semp\_demo.c  /\* 使用osal接口需要包含该头文件 \*/  #include <osal.h>  /\* 任务优先级宏定义（shell任务的优先级为10） \*/  #define USER\_TASK1\_PRI  12  //低优先级  #define USER\_TASK2\_PRI  11  //高优先级  /\* 信号量索引ID \*/  osal\_semp\_t sync\_semp;  /\* 任务task1入口函数 \*/  static int user\_task1\_entry()  {      while(1)      {          /\* 在串口打印信息 \*/          printf("task 1 post a semp!\r\n");          /\* 打印完毕释放信号量 \*/          osal\_semp\_post(sync\_semp);          /\* 任务主动挂起2s \*/          osal\_task\_sleep(2\*1000);      }  }  /\* 任务task2入口函数 \*/  static int user\_task2\_entry()  {      while (1)      {          /\* 优先级高，抢占执行打印信息 \*/          printf("task2 is waiting for a semp...\r\n");          /\* 申请信号量，申请失败则挂起等待 \*/          osal\_semp\_pend(sync\_semp, cn\_osal\_timeout\_forever);          /\* 一旦申请到信号量，则恢复执行 \*/          printf("task 2 access a semp!\r\n");      }  }  /\* 标准demo启动函数，函数名不要修改，否则会影响下一步实验 \*/  int standard\_app\_demo\_main()  {      /\* 创建信号量sync\_semp \*/      osal\_semp\_create(&sync\_semp, 1, 0);      printf("sync\_semp semp create success.\r\n");      /\* 创建任务task1 \*/      osal\_task\_create("user\_task1",user\_task1\_entry,NULL,0x400,NULL,USER\_TASK1\_PRI);      /\* 创建任务task2 \*/      osal\_task\_create("user\_task2",user\_task2\_entry,NULL,0x400,NULL,USER\_TASK2\_PRI);      return 0;  }  将编写的osal\_semp\_demo.c文件添加到makefile中，直接修改Demo文件夹下的user\_demo.mk配置文件  #example for osal\_semp\_demo      ifeq ($(CONFIG\_USER\_DEMO), "osal\_semp\_demo")          user\_demo\_src  = ${wildcard $(TOP\_DIR)/targets/STM32L431\_BearPi/Demos/osal\_kernel\_demo/osal\_semp\_demo.c}          user\_demo\_defs = -D CONFIG\_OSAL\_SEMP\_DEMO\_ENABLE=1  endif  4.实验现象  779b41ff12e2c1a8911ebd83fb8419a  串口内容：  **linkmain:V1.2.1 AT 13:30:59 ON July 6 2021**  **sync\_semp semp create success.**  **WELCOME TO IOT\_LINK SHELL**  **LiteOS:/>task2 is waiting for a semp...**  **task 1 post a semp!**  **task 2 access a semp!**  **task2 is waiting for a semp...**  **task 1 post a semp!**  **task 2 access a semp!**  **task2 is waiting for a semp...**  **task 1 post a semp!**  **task 2 access a semp!** | | | |
| **【**教师评语和成绩**】**  **成绩：** **指导教师：** **日期：** | | | |